

(/) Platforms (/explore) Projects (/projects) Contest (/contest2020)

Cool weather station on screen

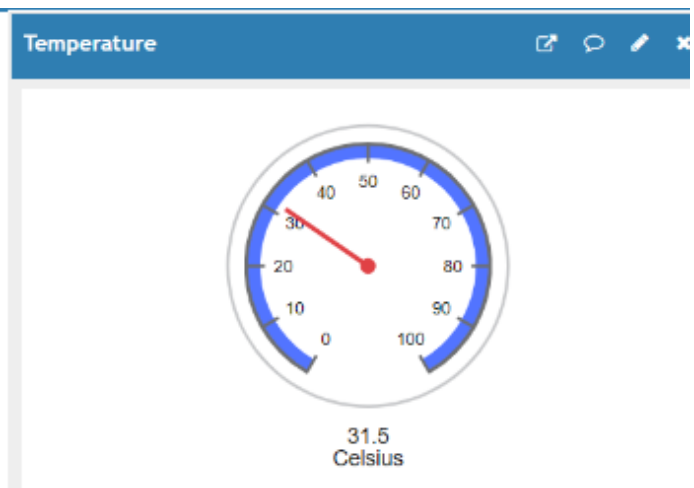
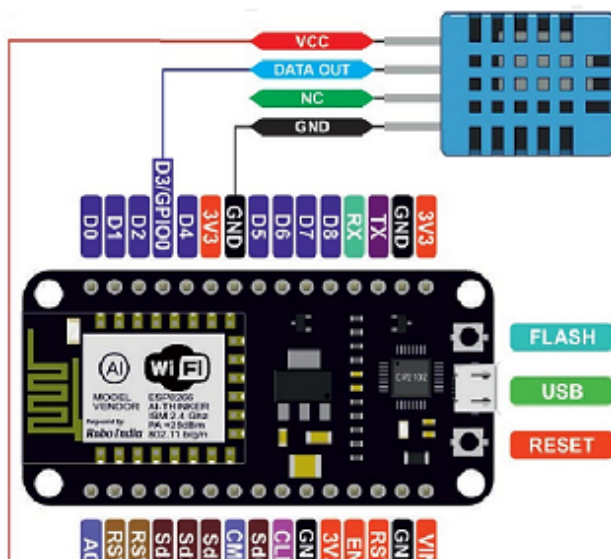
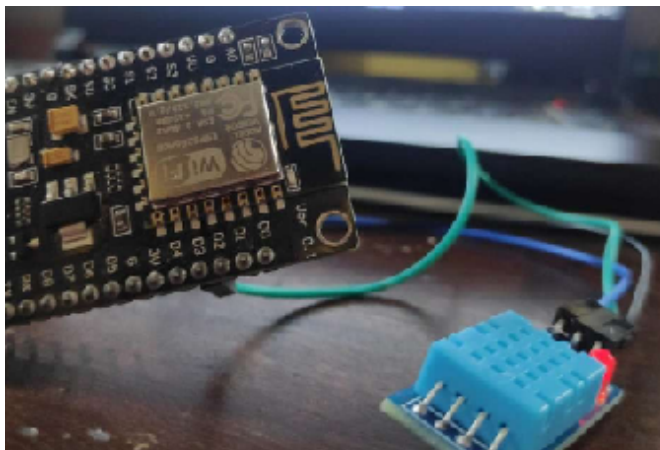


Published Jul 24, 2020 © [GPL-3.0-only](https://opensource.org/licenses/GPL-3.0) (<https://opensource.org/licenses/GPL-3.0>)

🕒 1.5 hours to build 🧑 Beginner

[nodemcu \(/search?query=nodemcu\)](/search?query=nodemcu) [arduino-esp8266 \(/search?query=arduino-esp8266\)](/search?query=arduino-esp8266)

We can easily get the weather data of the city we live in with the help of smartphones. But what if we can know the temperature and humidity in our room and can visualise it as a graph? This project tells the temperature and humidity using a DHT11 sensor.



Components Used

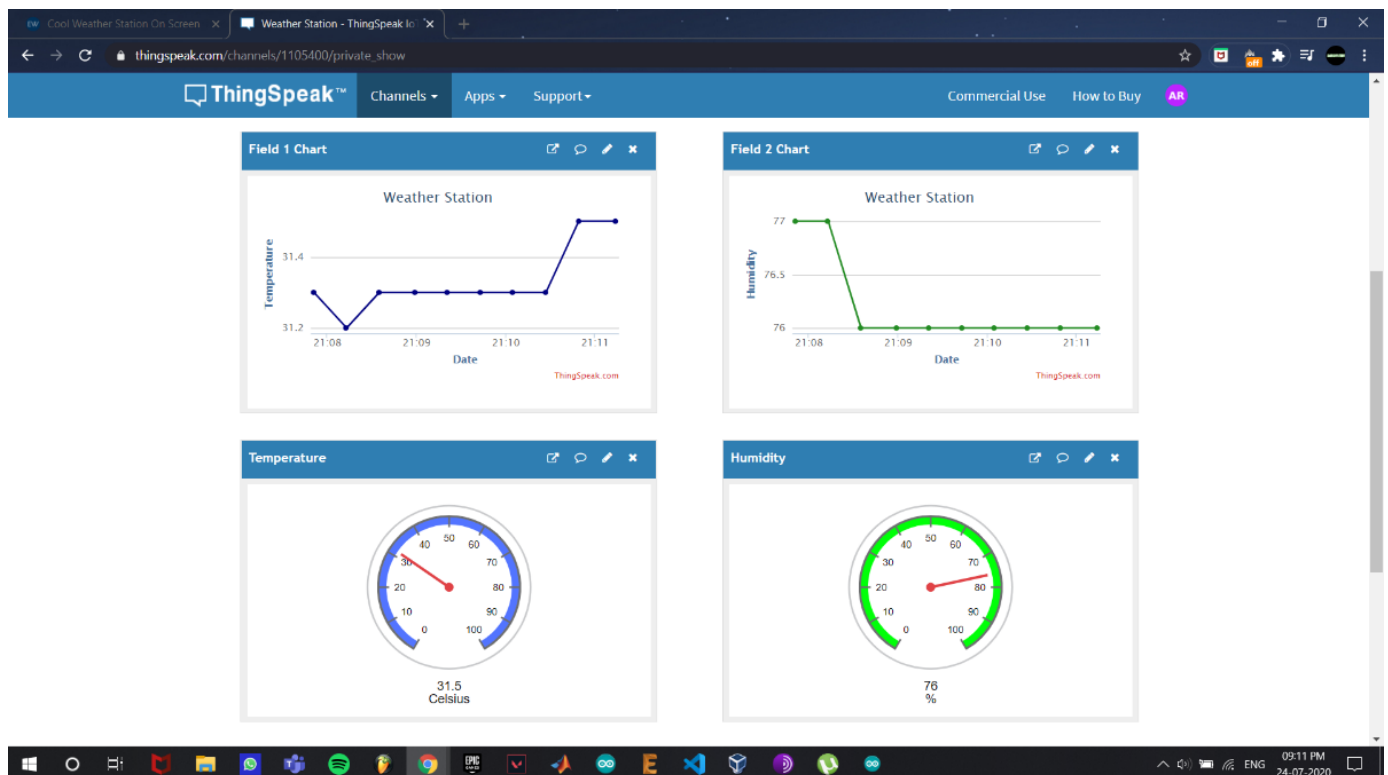
Components Used

[Add Project](#)
[Sign In](#)
[\(/login#login\)](#)

- 1 **DHT11** [Platforms \(/explore\)](#) [Projects](#)
[\(/components/dht11\)](#) [\(/projects\)](#) [Contest \(/contest2020\)](#)
 DHT11 is a single Wire digital humidity and temperature sensor, which gives relative humidity in percentage and temperature in degree Celsius.
- 1 **Node Mcu (Esp8266)**
 This will work as the bridge to send the data to thingspeak from DHT11 sensor module
- 1 **Wires**
 as per connections requirement
- 1 **USB cable**
 cable need for connection of nodemcu esp8266 and laptop

Description

The weather station project uses an esp8266 Nodemcu microcontroller board and a DHT11 temperature and relative humidity sensor module. The microcontroller receives the data from the DHT11 sensor and acts as a bridge in sending the data to the Thingspeak website, which I have used to plot the temperature and relative humidity as a graph and also on a gauge.



Visualisation of the results

The connections are easy as well. The wires go as follows:

1. The VCC pin of DHT11 goes to the 3V pin of the Nodemcu.
2. The GND pin of DHT11 goes to the GND pin of Nodemcu.
3. The data pin goes to GPIO0 (D3) of Nodemcu.

(//)

Platforms (/explore)
(/projects)

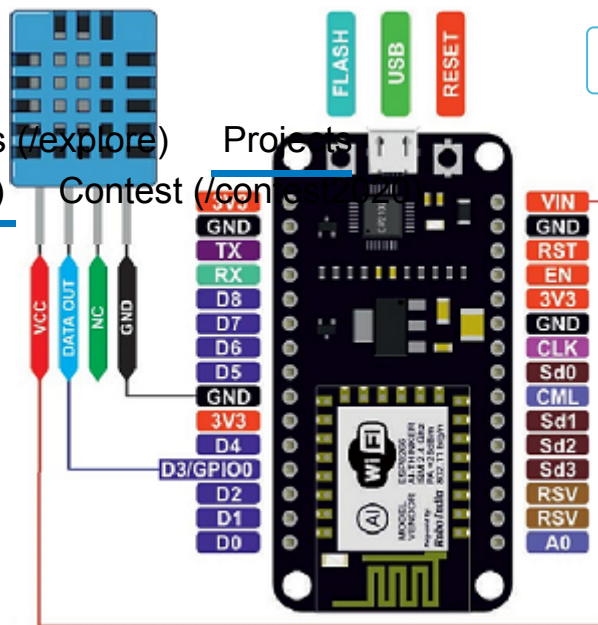
Contest (/con)

Projects (/con)

Add Project

Sign In

(/login#login)



The connections are done. Connect the microcontroller to the laptop or PC and then the microcontroller is programmed.

The DHT11 sensor gives the relative humidity and temperature data. The Nodemcu is programmed using the Arduino IDE so as to receive the data. Then the Nodemcu uses the HTTP POST request method to communicate with the Thingspeak website. The Thingspeak account has unique API keys which is used to communicate with the website.

Channel Settings

Percentage complete: 50%

Channel ID: 1105400

Name: Weather Station

Description: Temperature and Humidity data

Field 1: Temperature ☒

Field 2: Humidity ☒

Field 3: ☐

Field 4: ☐

Field 5: ☐

Field 6: ☐

Field 7: ☐

Field 8: ☐

Help

Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.

Channel Settings

- Percentage complete:** Calculated based on data entered into the various fields of a channel. Enter the name, description, location, URL, video, and tags to complete your channel.
- Channel Name:** Enter a unique name for the ThingSpeak channel.
- Description:** Enter a description of the ThingSpeak channel.
- Field#:** Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
- Metadata:** Enter information about channel data, including JSON, XML, or CSV data.
- Tags:** Enter keywords that identify the channel. Separate tags with commas.
- Link to External Site:** If you have a website that contains information about your ThingSpeak channel, specify the URL.
- Show Channel Location:**
 - Latitude:** Specify the latitude position in decimal degrees. For example, the latitude of the city of London is 51.5072.
 - Longitude:** Specify the longitude position in decimal degrees. For example, the longitude of the city of London is -0.1275.
 - Elevation:** Specify the elevation position meters. For example, the elevation of the city of London is 35.052.

The channel settings of the website.

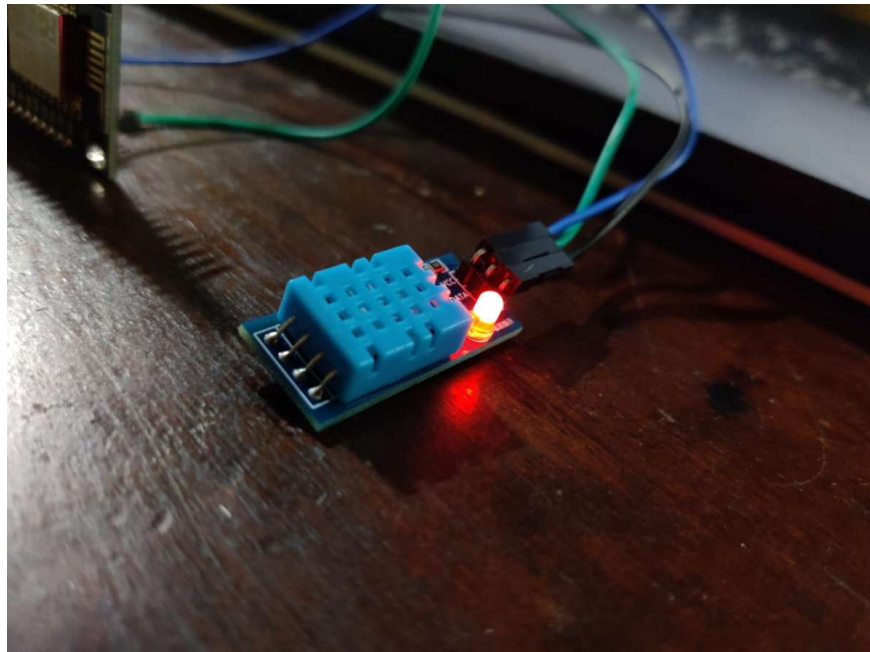
The HyperText Transfer Protocol acts works as a request-response protocol between the client and the server. Understanding how the HTTP protocol can be programmed will be very helpful in building the project. Generally, when using esp8266 Nodemcu board:

- The esp8266 Nodemcu acts as the client and submits an HTTP request to the server. In this project, the server corresponds to the Thingspeak website. [Add Project](#) [Sign In](#)
- The server returns a response to the client. [Platforms \(/explore\)](#) [Projects \(/login#login\)](#)
- finally, the response will contain the status information about the request and also might contain the content. Here the result is the graphs we see on Thingspeak. [Contest \(/contest2020\)](#)

The code has been added to the upcoming page.



ESP8266 Nodemcu Microcontroller



DHT11 sensor

Codes



Platforms (/explore) Projects (/login#login)
 Ambareeshan / weather_station_project (/users/Ambareeshan/codes/weather_station_project) (/projects) Contest (/contest2020) Project

The temperature and relative humidity are obtained using DHT11 sensor and plotted on thingspeak usin...

nodemcu (/search?query=nodemcu) arduino-esp8266 (/search?query=arduino-esp8266)

Attachments

Schematic

weather station project schematic 6 (/react/projects/download-attachment/272)

Credit



Ambareeshan Ramsharan (/users/Ambareeshan/profile)

1 Projects (/users/Ambareeshan/profile)

I am an electronics enthusiast more intrested in IoT, embedded systems and programming. I also have good knowledge in PCB designing.



1



50

Share ▾



Comments

Please login/signup to comment and reply.

Connect On: (https://www.facebook.com/electronicwings) (https://www.youtube.com/channel/UCNdqkukBtk4WhaBqcTsnn3w) (https://www.instagram.com/electronicwingscommunity/?hl=en)

About Us (/about) | Terms of Service (/terms-of-service) | © 2020 ElectronicWings
 Contact (/contactus)